

ВИКОРИСТАННЯ JAVA ДЛЯ РОЗРОБКИ МОБІЛЬНИХ ДОДАТКІВ: ПОРІВНЯННЯ З АЛЬТЕРНАТИВНИМИ ПЛАТФОРМАМИ

НТУ «Дніпровська політехніка»

Карапиш Ілля Володимирови

Наукові керівники: к.т.н., доц. Реута О.В., к.т.н., доц. Ширін А.Л.

Мобільні додатки стали необхідністю для бізнесу, розваг та соціального життя. Їх розробка може здаватися простим завданням, але це складний процес, який вимагає використання правильних інструментів. Прикладом такого інструменту є Java, одна з найпоширеніших мов програмування.

Java – це універсальна та потужна мова програмування, яка дозволяє розробникам створювати надійні та високопродуктивні додатки. Всупереч тому, що їй вже понад 25 років, вона все ще залишається вибором багатьох розробників для створення мобільних застосунків. Java є чудовим вибором для мобільної розробки, оскільки вона працює на більш ніж 6 мільярдах пристроїв і орієнтована на додатки для Android та iOS [1]. Основні особливості та переваги використання Java:

- **Об'єктно-орієнтованість:** як і Python, C++ та Ruby, Java є об'єктно-орієнтованою мовою програмування. Все є "об'єктом", який містить код і дані.

- **Незалежність від платформи:** програма на Java спочатку перетворюється в байт-код за допомогою компілятора Java, тому код може працювати на будь-якому комп'ютері, який підтримує середовище виконання Java і є незалежним від платформи.

- **Безпека:** Java набагато безпечніша за інші мови програмування, оскільки вона не залежить від платформи та має дуже мало взаємодії з операційною системою.

- **Потужність:** Java – це інтерпретована мова, але на відміну від C та C++, які компілюються, вона є дуже потужною завдяки власному компілятору just-in-time.

- **Багатопотоковість:** Java дозволяє писати додатки, які виконують декілька завдань в окремих потоках. Наприклад, Java-додаток може показувати користувачеві форму для входу в систему під час виконання фонових процесів.

- **Відкритий вихідний код:** протягом багатьох років Java накопичила велику колекцію бібліотек з відкритим вихідним кодом, які значно спрощують розробку мобільних додатків на Java.

- **Підтримка спільноти:** оскільки Java є традиційною мовою програмування, існує велика спільнота розробників, які діляться цінною інформацією та знаннями про розробку Java-додатків [2].

Попри всі переваги, у Java є певні недоліки, які можуть змусити розробників звернути увагу на інші платформи для мобільної розробки:

- **Продуктивність:** Java потребує інтерпретації під час виконання, що дозволяє їй працювати у кожній операційній системі, але це також робить її повільнішою, ніж такі мови, як C та C++.

- Споживання пам'яті: програма на Java споживає більше пам'яті, оскільки вона виконується з віртуальної машини Java.

- Вартість: мова програмування Java є коштовною через її високі вимоги до процесора та пам'яті. Розробникам потрібне краще обладнання для запуску програм на Java.

- Збір сміття: Java забезпечує автоматичне збирання сміття, яке не може контролюватися програмістом. У ній не передбачено таких методів, як delete() та free() для звільнення пам'яті [3].

Найвідомішими альтернативами Java для мобільної розробки є Flutter та React Native.

Flutter – це фреймворк для розробки мобільних додатків з відкритим вихідним кодом, створений компанією Google. Використовується для застосунків на Android та iOS.

Плюси Flutter:

- Flutter дозволяє миттєво вносити зміни в додаток, що є однозначним плюсом, коли мова йде про виправлення помилок.

- Додатки на основі Flutter працюють дуже плавно, що забезпечує чудовий UX.

- Завдяки єдиній кодовій базі, забезпечення якості та тестування зазвичай займає набагато менше часу.

- Розробка на Flutter дуже швидка та ефективна.

Мінуси Flutter:

- Додатки, створені за допомогою Flutter, як правило, мають великий розмір.

- Хоча Flutter популярний, він існує недостатньо довго, щоб мати велику ресурсну базу.

- Фреймворк Flutter побудований на мові Dart, яку доведеться вивчити додатково, щоб працювати з ним [4].

React Native - це популярний кросплатформний фреймворк для розробки мобільних додатків, створений компанією Facebook, який дозволяє розробникам створювати додатки для платформ iOS та Android. Як і будь-яка технологія, React Native має свої сильні та слабкі сторони. Ось переваги та недоліки використання React Native для розробки мобільних додатків.

Переваги:

- Багаторазове використання коду: React Native дозволяє розробникам написати код один раз і використовувати його на платформах iOS та Android.

- Нативна продуктивність: розробники можуть створювати додатки, які досягають такого ж рівня продуктивності, як нативні.

- Гаряче перезавантаження: React Native має функцію гарячого перезавантаження, яка дозволяє програмістам виправляти свій код і бачити зміни в реальному часі.

- Велика спільнота: для розробників доступна широка підтримка, включаючи бібліотеки, документацію та форуми.

- Економічна ефективність: React Native дозволяє використовувати єдину кодову базу для створення додатків для iOS та Android.

Недоліки:

- Обмежений нативний доступ: розробники можуть зіткнутися з деякими обмеженнями при створенні додатків, які потребують доступу до певних нативних API.

- Обмеження продуктивності: React Native дозволяє створювати додатки, які поведуться як нативні, але можуть мати обмеження продуктивності при створенні складних застосунків.

- Відсутність кастомізації: React Native використовує попередньо зібрані компоненти, які не можуть бути кастомізовані тією мірою, в якій цього вимагають деякі додатки.

В цілому, Java вже давно є популярним вибором для розробки мобільних застосунків завдяки своїй зрілості та універсальності. Однак з появою нових технологій розробники почали розглядати альтернативні платформи, такі як Flutter та React Native. Flutter популярний тим, що дозволяє створювати адаптивні та візуально привабливі додатки за короткий час. Він також надає функцію гарячого перезавантаження, що пришвидшує процес розробки. React Native, з іншого боку, забезпечує швидшу розробку, дозволяючи розробникам писати код на JavaScript і застосовувати його на платформах iOS та Android. Не дивлячись на свої переваги, Flutter і React Native мають деякі обмеження. Flutter все ще є відносно новою платформою і може не мати деяких функцій, до яких розробники звикли при використанні більш усталених технологій мови Java. React Native пропонує крос-платформну розробку, але може бути складним у налаштуванні та вимагати додаткових ресурсів для підтримки.

Таким чином, вибір правильної платформи для розробки мобільних додатків залежить від розробників та конкретних потреб проекту. Щоб приймати обґрунтовані рішення розробники повинні враховувати такі фактори як час, бюджет, функціональність та цільова аудиторія.

Перелік посилань

1. Yulia Borusiuk. Benefits of Java for Mobile Application Development. 2022 [Електронний ресурс]. URL: <https://ncube.com/blog/benefits-of-java-for-mobile-application-development> (дата звернення: 10.04.2023).

2. Java Mobile Applications Development: What You Need to Know. 2021 [Електронний ресурс]. URL: <https://www.webiotic.com/java-mobile-applications-development-what-you-need-to-know/> (дата звернення: 10.04.2023).

3. Disadvantages of Java language. Graphic Era University (GEU) - 2019 [Електронний ресурс]. URL: <https://www.geeksforgeeks.org/disadvantages-of-java-language/> (дата звернення: 10.04.2023).

4. Maciek Berka. Flutter Pros & Cons – Should You Use It In Your Project? 2020 [Електронний ресурс]. URL: <https://invotech.co/blog/flutter-pros-cons-should-you-use-it-in-your-project/> (дата звернення: 10.04.2023).